

1. Einleitung

- 1.1 Du kennst den Namen des Informatikers, der die Programmiersprache „Python“ entwickelt hat
- 1.2 Du kannst beschreiben, worauf sich der Name „Python“ bezieht.
- 1.3 Du kannst angeben, welche Art von Programmen nötig sind, um ein Python-Programm zu schreiben und um ein Python Programm auszuführen.
- 1.4 Du kennst das Zeichen, das Python für Kommentare verwendet.
- 1.5 Du kannst die drei Fehlerkategorien beim Programmieren nennen und jeweils ein Beispiel angeben.

2. Arithmetik

- 2.1 Du kannst Arithmetische Ausdrücke mit folgenden Operatoren auswerten:
 - Grundrechenarten (+, -, *, /)
 - Modulo-Operator (%)
 - Ganzzahl-Division (//)
 - Potenzen und Wurzeln (**)
- 2.2 Du kennst die Operator-Hierarchie (Klammern, Potenzen, „Punkt vor Strich“)
- 2.3 Du weisst, dass ohne Klammern die Operationen gleicher Stufe von links nach rechts ausgewertet werden. **Ausnahme:** Ohne Klammern werden beim Potenzieren die Operanden von rechts nach links ausgewertet. ($2**3**2$ ist 2^9 und nicht 8^2 .)
- 2.4 Du kennst die folgenden Datentypen
 - ganze Zahlen (`int`)
 - Fließkommazahlen (`float`)und kannst angeben, welchen Datentyp ein arithmetischer Ausdruck hat, wenn der Datentyp der Operanden bekannt ist.

3. Variablen

- 3.1 Du kannst gültige (und damit auch ungültige) Variablennamen erkennen. Als Python-Schlüsselwörter müssen nur diejenigen erkannt werden, die bisher im Unterricht behandelt wurden.
- 3.2 Du kannst Zeichenketten von Variablen unterscheiden.
- 3.3 Du kennst den Zuweisungsoperator „`=`“ und weisst, wie Zuweisungen ausgewertet werden.

- 3.4 Du kannst die erweiterten Zuweisungen (`+=`, `-=`, `*=`, `/=`, `//=`, `%=`, `**=`) interpretieren.
- 3.5 Du kannst Mehrfachzuweisungen interpretieren.
- 3.6 Du kannst Daten von der Shell (Konsole) mit der Funktion `input("...")` einlesen und weisst, dass als Ergebnis jeweils ein String (Zeichenkette) zurückgegeben wird.
- 3.7 Du kannst eine Zeichenkette, die eine Zahl darstellt, mit der Funktion `int(...)` bzw. `float(...)` in den entsprechenden Datentyp umwandeln.
- 3.8 Du verstehst die Semantik (Bedeutung) der `print(...)`-Anweisung. und weisst, dass auch mehrere, durch Kommas getrennte Argumente, nacheinander ausgegeben werden können.
- 3.9 Du kannst Programme schreiben, die ...
- Daten von der Shell einlesen,
 - Zeichenketten in den richtigen Datentyp umwandeln (siehe oben),
 - Berechnungen ausführen,
 - Ergebnisse ausgeben.

4. Wahrheitswerte

- 4.1 Du kannst die beiden Konstanten `True` und `False` des Python-Datentyps `Boolean` (`bool`) aufzählen und in Programmen richtig schreiben.
- 4.2 Du kannst Ausdrücke mit den Vergleichsoperatoren `<`, `<=`, `>`, `>=`, `==` und `!=` auswerten.
- 4.3 Du kannst Ausdrücke mit den logischen Operatoren `not`, `or`, `and` auswerten.
- 4.4 Du kennst die Hierarchie bei der Auswertung von logischen Ausdrücken, von Vergleichsausdrücken und von gemischten Ausdrücken mit arithmetischen Operatoren in Python:
1. `(...)`
 2. `**`
 3. `*`, `/`, `//`, `%`
 4. `+`, `-`
 5. `!=`, `==`, `<`, `<=`, `>`, `>=`
 6. `not(...)`
 7. `and`
 8. `or`

4.5 Du kannst Ausdrücke auswerten, in denen die Wahrheitswerte in entsprechende ganze Zahlen umgewandelt (*gecastet*) werden. *Beispiele:*

Ausdruck	Wert
<code>int(True)</code>	1
<code>int(False)</code>	0
<code>True + 2</code>	3
<code>10 * False</code>	0
<code>(5 < 7) + (3 != 4)</code>	2